Computing Machinery I
Assignment 3
8% of your final score
Due October 30th @ 11:59PM MST

Lead TA
The lead TA for this assignment is Lei Wang (lei.wang2@ucalgary.ca)

Objective
The objective of this assignment is to practice binary multiplication and shifting in ARMv8 assembly.

Skills Needed for this Assignment
- Ability to work with basic arithmetic, loops, and if-else constructs in assembly
- Ability to print to standard output using the `printf()`
- Ability to use shifting instructions to implement multiplication
- Ability to assemble programs using *gcc* and use *m4* to process macros
- Ability to use *gdb* to debug and display assembly language programs

Overview
Your programs will implement multiplication for binary numbers.

Details
Implement the following method to multiply two integers in ARM assembly:

0...0100 = 4 in binary 0...0011 = 3 in binary	Muliplier. Multipicand.
1*0...0100 = 0..00100	Least significant bit of multiplicand times multiplier.
1*0...0100 = 0..0100	2nd least significant bit of multiplicand times multiplier. Result shifted to the left one time.
0*0...0100 = 0..000	3nd least significant bit of multiplicand times multiplier Result shifted to the left one more time.
0*0...0100 = 0..00	4th least significant bit of multiplicand times multiplier Result shifted to the left one more time.
…	
0*0...0100 = 0	Most significant (64th) bit of multiplicand times multiplier. Result shifted to the left 63 times
= 0..01100 = 12	final result of multiplication is the sum of all 64 partial results

Your program should compute and display multiplicand, multiplier and result of the following multiplications using the above method.

-15 * random integer number between 0 and 15 = multiplication result
-14 * random integer number between 0 and 15 = multiplication result
...
-1 * random integer number between 0 and 15 = multiplication result
0 * random integer number between 0 and 15 = multiplication result
1 * random integer number between 0 and 15 = multiplication result
...
14 * random integer number between 0 and 15 = multiplication result
15 * random integer number between 0 and 15 = multiplication result

Note: mul/smul and madd/msub assembly instructions are not to be used.

Make sure your code is properly formatted into columns, is readable and fully documented, and includes identifying information at the top of each file. You must comment each line of assembly code. Your code should also be well designed: make sure it is well organized, clear, and concise.

Submission

- **Note:** The lead TA may provide further submission instructions
- Name your programs *assign3.asm*
- Create a script file for each program. Call them *assign3.script* **The script file must contain a GDB session.**
- Submit a *README* file providing extra instructions or information for your TA (optional)
- Submit your work to the appropriate dropbox on D2L.

Late Submission Policy

Late submissions will be penalized as follows:
-12.5% for each late day or portion of a day for the first two days
-25% for each additional day or portion of a day after the first two days
Hence, no submissions will be accepted after 5 days (including weekend days) of the announced deadline.

Academic Misconduct

This assignment is to be done by individual students: your final submission must be your own original work. Teamwork is not allowed. Any similarities between submissions will be further investigated for academic misconduct. While you are encouraged to discuss the assignment with your colleagues, this must be limited to conceptual and design decisions. Code sharing by any means is prohibited, including *looking* at someone else's paper or screen. The submission of compiler generated assembly code is absolutely prohibited. Any re-used code of excess of 5 lines in C and 10 lines in assembly (10 assembly language instructions) must be cited and have its source acknowledged. Failure to credit the source will also result in a misconduct investigation.

D2L Marks

Marks posted on D2L are subject to change (up or down).

Computing Machinery I
Assignment 3 Rubric

Student:____

Item	**Max Points**	**Points**
Code compiles	5	
Code runs	5	
Correct results displayed	25	
Multiplication implemented as described	40	
Script file	10	
Code readability (formatting and documentation)	15	
Total Points	**100**	